

Cybersecurity

2022/2023

Solutions of the exercises

- 1) Let us consider a simple monoalphabetic shift cipher (Caesar's Cipher), with an alphabet of N characters (with $N=26$), with a secret key $K=4$ (the shift). Do encrypt the text "SECRET".

SOLUTION

A shift of $K=4$ leads to the following encryption substitution table:

Cleartext char: abcdefghijklmnopqrstuvwxyz

Ciphertext char: EFGHIJKLMNOPQRSTUVWXYZABCD

then:

Cipher text: $c = E_k(m) = \text{SHIFT}(4, \text{"SECRET"}) = \text{"WIGVIX"}$

- 2) Consider a monoalphabetic substitution cipher, that maps a plaintext character M into the cipher character C , defined as follows:

$$C = E_k(M) = aM + b \bmod 26$$

where M is any character of the alphabet $\{'a', 'b', 'c', \dots, 'z'\}$, and a and b are two integer parameters that form the secret key $K = \langle a, b \rangle$

By using such a cipher, a ciphertext has been generated starting from an English plaintext. By analyzing the ciphertext it results that the most frequent letter of the ciphertext is 'B', and the second most frequent letter of the ciphertext is 'U'.

Try to break this code, by knowing that the two most frequent letters in English are 'e' and 't'.

(Hints: $x \bmod n = y \Rightarrow \exists h : x = y + hn$. The equation $15x \bmod 26 = 19$ has the solution $x = 3$).

SOLUTION

$$M1 = 'e' = 4 \rightarrow C1 = 'B' = 1$$

$$M2 = 't' = 19 \rightarrow C1 = 'U' = 20$$

$$(4a+b) \bmod 26 = 1$$

$$(19a+b) \bmod 26 = 20$$

$$\exists h : b = 1 - 4a + h \cdot 26$$

$$(19a - 4a + 1 + h \cdot 26) \bmod 26 = 20$$

$15a \bmod 26 = 19$, that has the solution $a=3$ (to find the solution, you can use the Euclid's algorithm to find 15^{-1} modulo 26, that is 7; then by multiplying both sides by 7 you obtain $a = 7 \cdot 19 \bmod 26 = 3$)

Then, $a = 3$, and $b = 1 - 4 \cdot 3 + h \cdot 26 = 15$

$$C = E_k(M) = 3M + 15 \bmod 26$$

- 3) Starting from a block cipher $E_K(\cdot)$ with block size q , please show the scheme for the CBC (Cipher Block Chaining) encryption of a message m with length $L > q$ (for simplicity, let's consider $L = nq$).

SOLUTION

If we express m and c as:

$$m = M1 \parallel M2 \parallel \dots \parallel M_n$$

$$c = C_1 || C_2 || \dots || C_n$$

it is:

$$C_0 = IV$$

$$C_i = E_K(M_i \oplus C_{i-1})$$

- 4) Suppose to have an API implementing a block cipher E in CBC mode, with block size q . The same block cipher in CBC mode has been used to encrypt a message m with length pq using a key K of size n bits. Evaluate the complexity of a brute force attack against the secret key K , by supposing to know both the plaintext m and the ciphertext c . In each attempt, the entire message is processed. Indicate the complexity in terms of the number of block encryptions (using the function E), as function of n , p and q .

SOLUTION

Given the message m , the maximum number of keys that should be tried (worst case) in order to find the right key K such that $E\text{-CBC}(K, m) \equiv c$ is 2^n . Since each attempt requires the execution of p encryption operations, the complexity of this attack in terms of number of E operations is:

$$p 2^n.$$

If T_E is the time for one encryption with $E(\cdot)$, the total time required for the complete brute-force attack is: $p 2^n T_E$

The same result could be obtained by using the decryption function $D\text{-CBC}(K, c)$ and searching the key K such that: $D\text{-CBC}(K, c) \equiv m$.

- 5) Let us consider a symmetric block cipher $E_k(\cdot)$ with size 4 bit.
By supposing that, given a secret key K , the encryption table of $E_k(\cdot)$ corresponds to the table at the right side, do encrypt in CBC mode with $IV=0000$ the following plaintext message:

$$m = 1100 \ 1010 \ 0010 \ 1101$$

| <i>plaintext</i> | <i>ciphertext</i> |
|------------------|-------------------|
| 0000 | 1110 |
| 0001 | 0100 |
| 0010 | 1101 |
| 0011 | 0001 |
| 0100 | 0010 |
| 0101 | 1111 |
| 0110 | 1011 |
| 0111 | 1000 |
| 1000 | 0011 |
| 1001 | 1010 |
| 1010 | 0110 |
| 1011 | 1100 |
| 1100 | 0101 |
| 1101 | 1001 |
| 1110 | 0000 |
| 1111 | 0111 |

SOLUTION

Encryption is performed in CBC mode, that is:

$$C_i = E_k(M_i \text{ XOR } C_{i-1})$$

$$\text{with } C_0 = IV = 0000$$

then:

$$C_1 = E_k(1100 \text{ XOR } 0000) = E_k(1100) = 0101$$

$$C_2 = E_k(1010 \text{ XOR } 0101) = E_k(1111) = 0111$$

$$C_3 = E_k(0010 \text{ XOR } 0111) = E_k(0101) = 1111$$

$$C_4 = E_k(1101 \text{ XOR } 1111) = E_k(0010) = 1101$$

$$c = 0101 \ 0111 \ 1111 \ 1101 \quad (iv=0000)$$

- 6) Let us consider the following plaintext message:

$$m = 1100 \ 0000 \ 1100 \ 0000$$

encrypted by means of the same symmetric encryption algorithm $E_k(\cdot)$ with block size 4bit and secret key K of the previous exercise (same encryption/substitution table) in OFB mode with $IV=0001$, resulting the following ciphertext:

$$c = 1000 \ 0010 \ 0001 \ 1001 \quad (IV=0001)$$

Show how it is possible to modify the ciphertext c in such a way that by decrypting it you obtain the following plaintext:

$$m' = 1100 \ 0000 \ 1001 \ 0000$$

SOLUTION

Encryption has been done in OFB mode, that is $c = m \oplus o$.

Hence, if you want to change a bit of the decrypted plaintext you have to change the corresponding bit of the ciphertext.

Referring to the third block:

original $M_3 = 1100$

target $M'_3 = 1001$

so you have to simply change the second and fourth bit of C_3 , that is:

original $C_3 = 0001$

modified $C'_3 = 0100$

$$c' = 1000 \ 0010 \ 0100 \ 1001 \ (iv=0001)$$

7) Let us consider a message $m = M1 \parallel M2 \parallel M3 \parallel M4$, and suppose to decrypt it by means of a block cipher $E_K()$ in CBC mode (the block size of $E_K()$ is equal to the size of the blocks M_i), with $iv = IV0$, obtaining the ciphertext $c = C1 \parallel C2 \parallel C3 \parallel C4$.

If an attacker modifies the ciphertext by rearranging the component blocks obtaining the new ciphertext $c' = C1 \parallel C3 \parallel C2 \parallel C4$, which will be the corresponding plaintext message $m' = M'1 \parallel M'2 \parallel M'3 \parallel M'4$ obtained by “erroneously” decrypting the ciphertext c' ? Show the blocks $M'i$ as function of Mj and Cj with $j=1..4$.

SOLUTION

With CBC encryption, it is:

$$C_i = E_K(M_i \oplus C_{i-1})$$

and:

$$M_i = D_K(C_i) \oplus C_{i-1}$$

and also:

$$D_K(C_i) = M_i \oplus C_{i-1}$$

indicating with:

$$m' = M'1 \parallel M'2 \parallel M'3 \parallel M'4$$

by setting:

$$c' = C1 \parallel C3 \parallel C2 \parallel C4$$

it results:

$$M'1 = D_K(C'1) \oplus IV0 = D_K(C1) \oplus IV0 = M1$$

$$M'2 = D_K(C'2) \oplus C'1 = D_K(C3) \oplus C1 = (M3 \oplus C2) \oplus C1$$

$$M'3 = D_K(C'3) \oplus C'2 = D_K(C2) \oplus C3 = (M2 \oplus C1) \oplus C3$$

$$M'4 = D_K(C'4) \oplus C'3 = D_K(C4) \oplus C2 = (M4 \oplus C3) \oplus C2$$

8) Realize a symmetric encryption scheme for encrypting messages m with any length, based on a block cipher $E_K()$ (e.g. AES), without obtaining avalanche effect, in such a way that if you change one bit of the ciphertext, only one bit of the plaintext will change when decrypting the ciphertext (hint: use the XOR operator).

SOLUTION

$$m = M1 \parallel M2 \parallel \dots \parallel M_n$$

$$c = IV \parallel C1 \parallel C2 \parallel \dots \parallel C_n$$

$$C_i = M_i \oplus O_i$$

with:

$$O_i = E_K(O_{i-1}) = \text{AES}(K, O_{i-1})$$

$$O_0 = IV$$

- 9) Consider the following three padding algorithms for extending the length of a message to a multiple of N bytes (e.g. $N=32$). Which of the three algorithms are suitable for using with a block cipher with block size N bytes? Why?
- Padding1: append to the message random bytes until the total length (in bytes) becomes a multiple of N .
- Padding2: append to the message random bytes until the total length (in bytes) becomes a multiple on $N - 1$; append one byte encoding the number of padding bytes that have been added.
- Padding3: append to the message a bit '1', then append as many bits '0' as needed to reach a multiple of N bytes.

SOLUTION

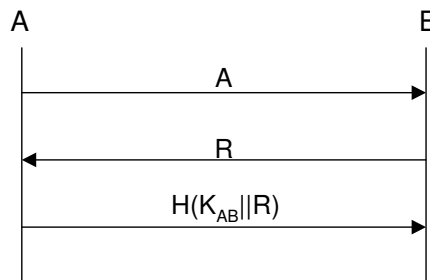
All three padding algorithms extend the message length to a multiple of N . However only Padding2 and Padding3 are suitable for encryption/decryption, since they allow the receiver to detect the end of the original message and to correctly remove the padding data after decryption.

- 10) Starting from a hash function $H()$ and a symmetric key K_{AB} shared between two entities A and B :

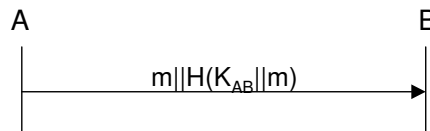
- show a possible authentication scheme between A (supplicant) and B (authenticator);
- show how it is possible to send a message m from A to B providing data authentication and integrity protection;
- create an encryption function (and the corresponding decryption function) that can be used for sending a message m encrypted from A to B .

SOLUTION

- i) A possible authentication scheme between A (supplicant) and B (authenticator):



- ii) Authentication and integrity protection of a message m sent from A to B :



- iii) Encryption function (and the corresponding decryption function) that can be used for encrypting the message m from A to B :

Let's define

$$O_0 = IV$$

$$O_i = H(K_{AB} || O_{i-1})$$

$$o = O_1 || O_2 || O_3 || \dots || O_n || \dots$$

$$c = E(K_{AB}, IV, m) = m \oplus o$$

message that is sent:

$$A \rightarrow B : x = IV || c$$

decryption:

$$m = E(K_{AB}, IV, c) = c \oplus o$$

- 11) Find the multiplicative inverse of each nonzero element in Z_7 .

SOLUTION

$$Z_7^* = \{1, 2, 3, 4, 5, 6\}$$

Corresponding multiplicative inverses: 1, 4, 5, 2, 3, 6

12) Find all nonzero elements in Z_{21} that are relatively prime with 21.

SOLUTION

Elements in Z_{21} that are co-prime with 21 are: $U_{21} = \{1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20\}$.

Note that: $\Phi(21) = \Phi(3 \times 7) = (3-1)(7-1) = 12 = |U_{21}|$

13) By using the Euclid's algorithm, find the greatest common divisor $\gcd(,)$ of:

- a) 36, 15
- b) 47, 20
- c) 43, 35

SOLUTION

a) $\gcd(36, 15) = (36, 15) = (15, 6) = (6, 3) = 3$

b) $\gcd(47, 20) = (20, 7) = (7, 6) = (6, 1) = 1$

c) $\gcd(43, 35) = (35, 8) = (8, 3) = (3, 2) = (2, 1) = 1$

14) Prove the following: If p and q are prime, then $\Phi(pq) = (p-1)(q-1)$.

(Hint: What numbers have a factor in common with pq ?)

SOLUTION

The integers that are less than pq and have a factor in common with pq are: $p, 2p, 3p, \dots, (q-1)p, q, 2q, 3q, \dots, (p-1)p$

In total they are $(q-1) + (p-1)$ values.

Since the total number of values less than pq is: $pq-1$,

then:

$$\Phi(pq) = pq - 1 - [(q-1) + (p-1)] = pq - p - q + 1 = (p-1)(q-1)$$

15) Create a pair of public/private RSA keys, using as p and q primes the values $p=3, q=11$. With such keys, do encrypt the plaintext message $m=2$.

SOLUTION

$$n=pq=33$$

$$\phi(n)=(p-1)(q-1)=20$$

Possible values for e and d are: 1, 3, 7, 9, 11, 13, 17, 19 (co-primes of 20)

If we choose $e=7$

using the extended Euclid's algorithm:

| | | |
|----|----|----|
| 20 | 1 | 0 |
| 7 | 0 | 1 |
| 6 | 1 | -2 |
| 1 | -1 | 3 |

that gives $d=3$, with $ed=1 \bmod \phi(n)$

If we define the public and private keys as: $K^+ = \langle e, n \rangle$ and $K^- = \langle d, n \rangle$

By encrypting m with the public key K^+ we have:

$$c=E(m)=2^7 \bmod 33=29$$

it is also possible to verify that:

$$m=D(c)=29^3 \bmod 33=((29 \cdot 29) \bmod 33) \cdot 29 \bmod 33=16 \cdot 29 \bmod 33=2$$

16) With the following values $p=7, q=11$ and $e=13$. Create a pair of public/private RSA keys $KU=\langle e, n \rangle$ and $KR=\langle d, n \rangle$ (Use the Euclid's algorithm for finding the value d). With such keys, do decrypt the ciphertext message $c=2$.

SOLUTION

$$n=77, \Phi(n)=60$$

$$e=13$$

By using the extended Euclid's algorithm:

| rk | ak | bk |
|----|----|-----|
| 60 | 1 | 0 |
| 13 | 0 | 1 |
| 8 | 1 | -4 |
| 5 | -1 | 5 |
| 3 | 2 | -9 |
| 2 | -3 | 14 |
| 1 | 5 | -23 |

that leads to:

$$1=5 \cdot 60 - 23 \cdot 13$$

that is:

$$(-23) \cdot 13 \equiv 1 \pmod{60}$$

$$d = e^{-1} = (-23) = 37$$

Then:

$$m = 2^{37} \pmod{77} = 51$$

Verify:

$$51^{13} \pmod{77} = 2 = c$$

- 17) In a public-key system using RSA, you intercept the ciphertext $C = 10$ sent to a user whose public key is $e = 5$, $n = 35$. What is the plaintext M ?

SOLUTION

It is easy to find that $n = 35 = 5 \times 7$, then $p=5$, $q=7$, $\Phi(pq)=24$.

Since it is: $d \cdot e \equiv 1 \pmod{24}$,

then $d=5$, and $M = 10^d \pmod{n} = 10^5 \pmod{35} = 5$

- 18) In an RSA system, the public key of a given user is $e = 31$, $n = 901$. What is the private key of this user?
(Hint: First use trial-and-error to determine p and q ; then use the extended Euclidean algorithm to find d)

SOLUTION

By trying to divide $n=901$ by different prime p values, we find $p=17$, and $q = n/p = 53$.

Hence, $\Phi(n) = 16 \times 52 = 832$, and (by using the Euclid's algorithm) $d = e^{-1} \pmod{832} = 671$.

- 19) Show an example of shared key exchange between A and B based on Diffie-Hellman scheme, using the generator $g=2$ and the prime $p=11$.

SOLUTION

If A chooses the secret $x_a=5$, while B chooses the secret $x_b=3$, we have (Diffie-Hellman exchange):

A send to B: $ya = g^{x_a} \pmod{p} = 10$

B send to A: $yb = g^{x_b} \pmod{p} = 8$

starting from ya and xb , B computes: $K_{BA} = ya^{x_b} \pmod{p} = 10^3 = 100 \times 10 = 1 \times 10 = 10$

starting from yb and xa , A computes: $K_{AB} = yb^{x_a} \pmod{p} = 8^5 = (8^2)^2 \times 8 = 9^2 \times 8 = 4 \times 8 = 10$

with: $K_{AB} = K_{BA}$

- 20) Show that 2 is a primitive root of 11.

SOLUTION

By computing $g^1, g^2, \dots, g^k \bmod 11$, with $g=2$, we obtain: 2,4,8,5,10,9,7,3,6,1, that are all nonzero elements in Z_{11} that are coprime with 11 (since 11 is prime, all nonzero integer less than 11 are coprime with 11); that means that 2 is a primitive root of 11.

Alternatively:

From the previous computed values, it is possible to see that the first m such as $g^m = 1 \bmod 11$, is $m=10=\Phi(11)$.

- 21) Users A and B use the Diffie-Hellman key exchange technique with a common prime $p=71$ and a primitive root $g=7$.
- If user A has private key $x_A=5$, what is A's public key y_A ?
 - If user B has private key $x_B=12$, what is B's public key y_B ?
 - What is the shared secret key K_{AB} ?

SOLUTION

$$y_A = 7^5 \bmod 71 = 51$$

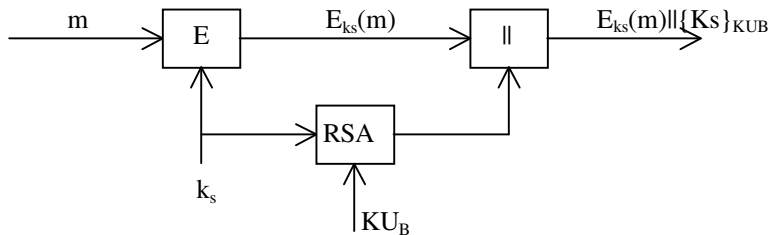
$$y_B = 7^{12} \bmod 71 = 4$$

$$K_{AB} = 4^5 \bmod 71 = 30 = 51^{12} \bmod 71 = 30$$

- 22) Let us suppose that you want to securely send a message m from A to B, by guaranteeing ONLY the data confidentiality. For message encryption you should use a symmetric encryption algorithm (since it is faster than asymmetric algorithm). By supposing that A and B share only their public RSA keys KU_A e KU_B (KR_A and KR_B are the private keys), show which functions can be executed at the sender and receiver sides. Try to depict the corresponding schemes.

SOLUTION

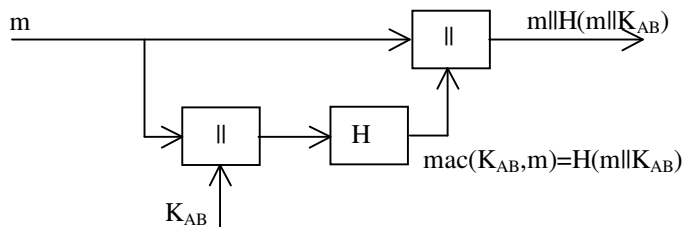
Sender:



- 23) Let us suppose that you want to securely send a message m from A to B, by guaranteeing ONLY data authentication/integrity. By supposing that A and B share only a secret key K_{AB} and a hash algorithm $H()$, show which functions can be executed at the sender and receiver sides. Try to depict the corresponding schemes.

SOLUTION

Sender:



- 24) Let us suppose that you want to securely send a message m from A to B, by guaranteeing both confidentiality and data authentication/integrity. For message encryption you should use a symmetric encryption algorithm (since it is faster than asymmetric algorithm). By supposing that A and B share only their public RSA keys KU_A e KU_B (KR_A and KR_B are the private keys), show which functions can be executed at the sender and receiver sides. Try to depict the corresponding schemes. A and B share the following algorithms: RSA, AES, SHA1.

SOLUTION

Data that are sent:

$$x = \text{AES}_{Ks}(m) \parallel \text{RSA}_{KU_B}(Ks) \parallel \text{RSA}_{KR_A}(H(m))$$

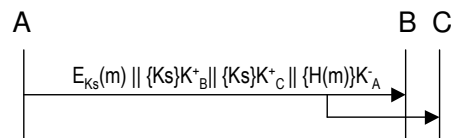
or also:

$$x = \text{AES}_{Ks}(m \parallel \text{RSA}_{KR_A}(H(m))) \parallel \text{RSA}_{KU_B}(Ks)$$

- 25) Let us suppose that you want to securely send a message m from A to two recipients B and C, by guaranteeing both confidentiality (through symmetric encryption with algorithm $E_k()$) and data authentication/integrity (through digital signature). Let us suppose that A, B and C have their own private RSA keys, K^-_A , K^-_B e K^-_C , and that they share all their public keys K^+_A , K^+_B e K^+_C . Please show which functions could be executed by A (sender), and the resulting message x that is actually sent from A to B and C.

SOLUTION

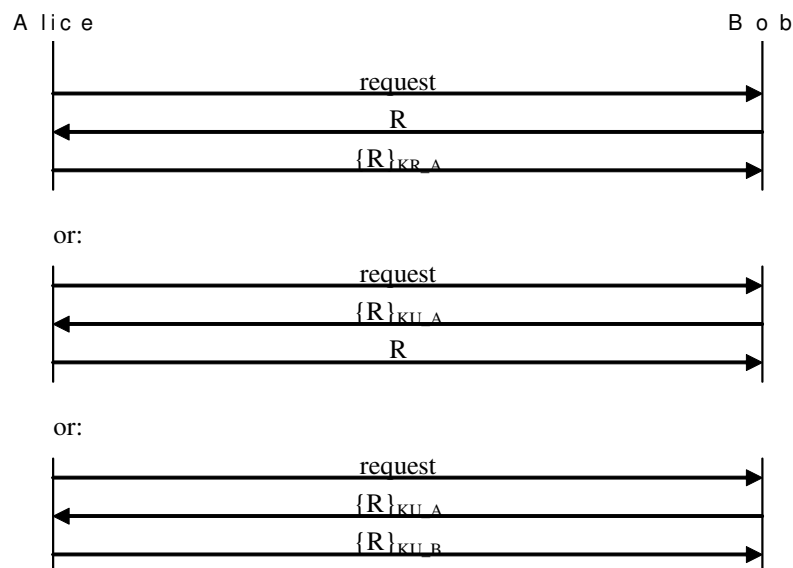
Sender:



Sent message: $x = E_{Ks}(m) \parallel \{Ks\}K^+_B \parallel \{Ks\}K^+_C \parallel \{H(m)\}K^-_A$

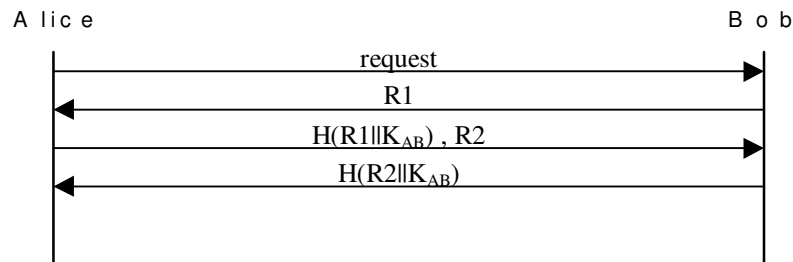
- 26) Show a possible secure authentication scheme between Alice (supplicant) and Bob (authenticator), by supposing that Alice and Bob share their public RSA keys KU_A and KU_B (KR_A and KR_B are the corresponding private keys).

SOLUTION



27) Show a possible mutual authentication scheme between Alice and Bob, based on the use of an hash function $H(\cdot)$ and a shared secret K_{AB} .

SOLUTION



28) Show a possible key transport scheme between two entities A and B, based on asymmetric encryption (public key cryptography), without the use of a KDC.

SOLUTION

A possible key distribution scheme between A and B is:

$A \rightarrow B: \{Ks, \text{sign}_A(\text{ID}_B, Ks)\}_{K_{Ub}}$

This scheme guarantees implicit key authentication to A, key authentication and confirmation to B, but no key confirmation to A. It also doesn't guarantee key freshness to B.

In order to add key freshness guarantee (to B), a timestamp can be also included:

$A \rightarrow B: \{Ks, t, \text{sign}_A(\text{ID}_B, Ks, t)\}_{K_{Ub}}$

29) Show an example of authenticated DH exchange that holds out against MITM attack.

SOLUTION

An example of authenticated DH that uses only digital signature is:

$A \rightarrow B: A, g^{Xa}$

$A \leftarrow B: B, g^{Xb}, \text{Sign}_B(g^{Xa} \parallel g^{Xb} \parallel A)$

$A \rightarrow B: \text{Sign}_A(g^{Xa} \parallel g^{Xb} \parallel B)$

An authenticated DH that uses both signature and encryption is (it is a variant of the STS protocol):

$A \rightarrow B: g^{Xa}$

$A \leftarrow B: g^{Xb}, E_{Ks}(B \parallel \text{Sign}_B(g^{Xa} \parallel g^{Xb}))$

$A \rightarrow B: E_{Ks}(A \parallel \text{Sign}_A(g^{Xa} \parallel g^{Xb}))$

Where K_s is a key derived by the DH result g^{XaXb} .

30) Let us consider an entity A that holds the following digital certificates: $\text{cert}_{CA3}(A)$, $\text{cert}_{CA2}(CA3)$, $\text{cert}_{CA1}(CA2)$, and $\text{cert}_{CA1}(CA1)$ (where $\text{cert}_Y(X)$ refers to the certificate of X signed by Y). Indicate what A should send to B in order to let A and B start a secure communication, under the following different hypotheses:

SOLUTION

| | |
|--|--|
| B owns: | A should send to B: |
| $\text{cert}_{\text{CA1}}(\text{CA1})$ | $\text{cert}_{\text{CA3}}(\text{A}), \text{cert}_{\text{CA2}}(\text{CA3}), \text{cert}_{\text{CA1}}(\text{CA2})$ |
| $\text{cert}_{\text{CA3}}(\text{A})$ | no additional certificate is required |
| $\text{cert}_{\text{CA1}}(\text{CA2})$ | $\text{cert}_{\text{CA3}}(\text{A}), \text{cert}_{\text{CA2}}(\text{CA3})$ |
| $\text{cert}_{\text{CA1}}(\text{CA1}), \text{cert}_{\text{CA3}}(\text{A})$ | no additional certificate is required |

31) If A holds $\text{cert}_B(\text{A})$ and $\text{cert}_C(\text{B})$ (where $\text{cert}_Y(\text{X})$ refers to the certificate of X signed by Y), while D holds $\text{cert}_E(\text{D})$, please indicate:

- what should A hold in order to authenticate D? Show a possible authentication scheme.
- what should D hold in order to authenticate A? Show a possible authentication scheme.

SOLUTION

a) what should A hold in order to authenticate D? Show a possible authentication scheme.

The public key of D,

OR the public key of E

In the latter case (A holds the public key of E), a possible authentication scheme is:

D → A: request

A → D: R

D → A: $\{\text{R}\}_{\text{KRd}}, \text{cert}_E(\text{D})$

Note: the $\text{cert}_E(\text{D})$ can be sent either in the first or in third message.

OR:

D → A: request, $\text{cert}_E(\text{D})$

A → D: $\{\text{R}\}_{\text{KUd}}$

D → A: R

b) what should D hold in order to authenticate A? Show a possible authentication scheme.

The public key of A,

OR the public key of B,

OR the public key of C.

In the latter case (D holds the public key of C), a possible authentication scheme is:

A → D: request

D → A: R

A → D: $\{\text{R}\}_{\text{KRd}}, \text{cert}_B(\text{A}), \text{cert}_C(\text{B})$

OR:

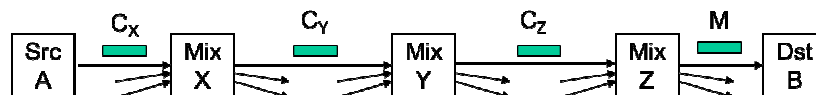
A → D: request, $\text{cert}_B(\text{A}), \text{cert}_C(\text{B})$

D → A: $\{\text{R}\}_{\text{KUa}}$

A → D: R

32) Let us consider an anonymizing network formed by high-latency anonymizing Mix nodes. Let us consider the case in which a node A wants to send a message m to a node B by means of three intermediate Mix nodes X, Y, and Z. Assume that K_i^+ and K_i^- are respectively the public and private keys of node i ($i=x,y,z$).

Indicate the format of the message C_X composed by A and sent to the first node X.



SOLUTION

Data sent by A to the first node X : $C_X = E_{K_X}^+ (ID_Y \parallel E_{K_Y}^+ (ID_Z \parallel E_{K_Z}^+ (ID_B \parallel m)))$

where ID_i is the identify or address of node i .

Note:

node X will receive such data, decrypt it with K_X and relay the following content data to Y : $E_{K_Y}^+ (ID_Z, E_{K_Z}^+ (ID_B, M))$

node Y will receive such data, decrypt it with K_Y and relay the following content data to Z : $E_{K_Z}^+ (ID_B, M)$

node Z will receive such data, decrypt it with K_Z and relay the message m to B .

- 33) Consider the following C function for verifying a user-provided password. Which type of attack it could be vulnerable to? What is a possible input password that could exploit such vulnerability?

```
int verifyPassword(char* pwd) {
    char str1[8];
    char str2[8];
    strcpy(str1, "SECRET"); // correct password is "SECRET"
    strcpy(str2, pwd);
    if (strncmp(str1, str2, 8) == 0) return 1; // compares the first 8 characters
    else return 0;
}
```

SOLUTION

It is vulnerable to buffer overflow attack. Any input string formed by a repetition of the same 8 characters may lead to a success.

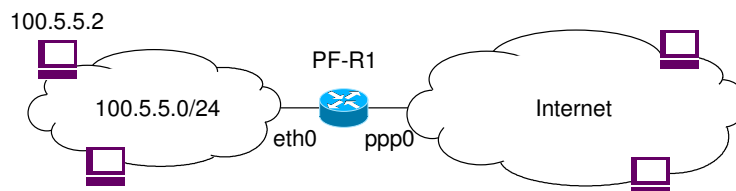
For example, input: aaaaaaabaaaaaab

When the input string is copied to `str2`, the second part 'aaaaaab' will overwrite `str1`. The result is that the comparison of the first 8 characters of `str1` and `str2` will match.

If the input string is much longer than 16 characters, the stack frame of the function can be damaged, resulting in a program error (possible DoS attack).

- 34) Let us consider the following network scheme, where in the node 100.5.5.2 there is a HTTP web server (TCP port 80) and a SMTP mail server (TCP port 25); you are requested to configure the filtering table of the router R1 so that:

- from external clients it is possible to access to the internal web server (node 100.5.5.2, TCP port 80);
- from internal clients it is possible to access any external web server (port 80);
- all client/server and server/client communications between the internal SMTP mail server and possible external SMTP servers are enabled; that is, internal SMTP Client → external SMTP Server (TCP port 25), and external SMTP Client → internal SMTP Server (TCP port 25).



SOLUTION

| FORWARD | | | | | | | | |
|--------------|---------------|--------------|-----------|-------|--------|--------|-------------|-----------------|
| Matching | | | | | | | | action |
| in_interface | out_interface | s_addr | d_addr | Proto | s_port | d_port | state | ACCEPT/ DROP |
| * | * | * | * | * | * | * | ESTABLISHED | ACCEPT |
| ppp0 | eth0 | * | 100.5.5.2 | TCP | * | 80 | NEW | ACCEPT |
| eth0 | ppp0 | 100.5.5.0/24 | * | TCP | * | 80 | NEW | ACCEPT |
| ppp0 | eth0 | * | 100.5.5.2 | TCP | * | 25 | NEW | ACCEPT |
| eth0 | ppp0 | 100.5.5.2 | * | TCP | * | 25 | NEW | ACCEPT |
| * | * | * | * | * | * | * | * | DROP |

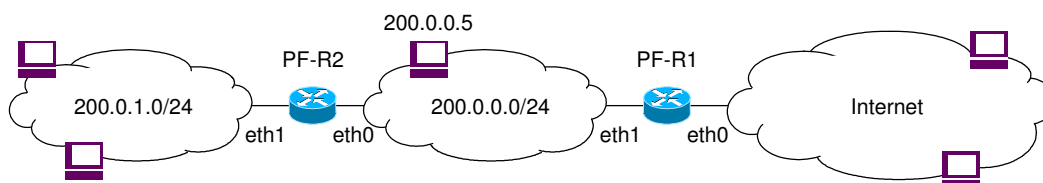
Or by applying anti-spoofing rules separately:

| FORWARD | | | | | | | | |
|--------------|---------------|--------------|-----------|-------|--------|--------|-------------|-----------------|
| Matching | | | | | | | | action |
| in_interface | out_interface | s_addr | d_addr | Proto | s_port | d_port | state | ACCEPT/ DROP |
| ppp0 | eth0 | 100.5.5.0/24 | * | * | * | * | * | DROP |
| * | * | * | * | * | * | * | ESTABLISHED | ACCEPT |
| * | * | * | 100.5.5.2 | TCP | * | 80 | NEW | ACCEPT |
| * | * | 100.5.5.0/24 | * | TCP | * | 80 | NEW | ACCEPT |
| * | * | * | 100.5.5.2 | TCP | * | 25 | NEW | ACCEPT |
| * | * | 100.5.5.2 | * | TCP | * | 25 | NEW | ACCEPT |
| * | * | * | * | * | * | * | * | DROP |

35) Let us consider the following company network formed by an internal network and a DMZ separated by a screening router R2, and connected to the external public network (Internet) through the screening router R1, as shown in figure.

You are requested to configure the filtering table of R1 so that:

- it is possible to establish application level client→server communications (through any transport protocol) from any DMZ node to any external node;
- it is blocked any attempt to establish a client→server communication from the external network to the DMZ;
- it is blocked any communication between the internal and the external networks;
- it is possible to establish TCP connections from the external network to the node 200.0.0.5 TCP port 80 (HTTP).



SOLUTION

| FORWARD | | | | | | | | |
|----------|---------|--------------|-----------|-------|--------|--------|-------------|-----------------|
| Matching | | | | | | | | action |
| in_int | out_int | s_addr | d_addr | Proto | s_port | d_port | state | ACCEPT/ DROP |
| * | * | * | * | * | * | * | ESTABLISHED | ACCEPT |
| eth1 | eth0 | 200.0.0.0/24 | * | * | * | * | NEW | ACCEPT |
| eth0 | eth1 | * | 200.0.0.5 | TCP | * | 80 | NEW | ACCEPT |
| * | * | * | * | * | * | * | * | DROP |